**Share My Books Application**

### Document Properties

| Item | Details |
| --- | --- |
| Document | Architecture Design Document |
| Application | Share My Books Application |
| Author | Veena Vijayan |
| Version | V1.0 |
| Creation Date | 22/10/2020 |
| Last Updated | 22/10/2020 |
| Status | Completed |

Contents

[Document Properties 1](#_Toc45989188)

[**1.** **Introduction** 3](#_Toc45989189)

[**2.** **Architecture Diagram** 3](#_Toc45989190)

[**3.** **Use-Case View** 4](#_Toc45989191)

[**4.** **Components** 4](#_Toc45989193)

[**5.** **Services** 5](#_Toc45989194)

[**6.** **Validations** 5](#_Toc45989195)

[**7.** **Tools and Libraries** 5](#_Toc45989196)

[**8.** **Unit Testing** 6](#_Toc45989197)

[**9.** **Responsive** 7](#_Toc45989198)

[**10.** **Accessibility** 7](#_Toc45989199)

[**11.** **LightHouse** 7](#_Toc45989199)

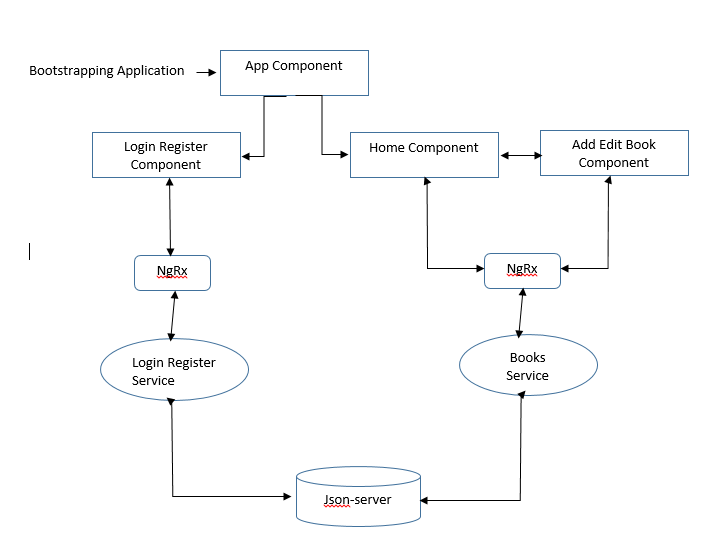
[**12.** **Storybook** 8](#_Toc45989199)

[**13.** **Setup** 8](#_Toc45989201)

# **Introduction**

This document provides a comprehensive architectural overview of Share My Books Application to depict different aspects and functionalities of the application.

# **Architecture Diagram**



# **Use-Case View**

The application use cases are –

* **Login –** It provides the login functionality to Share My Books Application.
* **Register –** It provides functionality to Register new user to Share My Books Application.
* **Lists Books** **–** Displays the List of books in home page.
* **Add Book –** It allows any logged in user to add new book details.
* **Update Book** **–** It allows any logged in user to update details of any book.
* **Delete Book** **–** It allows any logged in user to delete any book.
* **Theme** – It gives the option to toggle between Light and Dark theme.

# **Components**

* **AppComponent** – This component is used for Application Bootstrapping. Also**,** it includes the header, Login and Register buttons and theme selector..
* **Home Component** – This component contains the application header, Login/ Register/ Theming buttons.
* **AddEditBookComponent** – This component contains form and logic to add/edit single book.
* **LoginRegisterComponent** – This component contains form and logic for Login or Register functionality.

Login/Register and CRUD operations on book uses ngrx for state management.

# **Services**

* **AuthGuardService** – Implements CanActivate for routing user to dashboard page if not logged in.
* **BookService** **–** This service has the logic for getting books list, add new book details, update existing book details and delete entire book through respective API calls.
* **LoginRegisterService** – This service has logic for calling json-server-auth Login and Register API.
* **ThemeService** – This service has logic to toggle between Light and Dark themes.

# **Validations**

* Application has covered the required validation for Login/Register forms.
* Application shows proper error message for Login/Register failure i.e. incorrect email/password, duplicate email registration etc.

**Validation strategy:**

* Reactive form for validation in Login/Register
* Display error message received from service

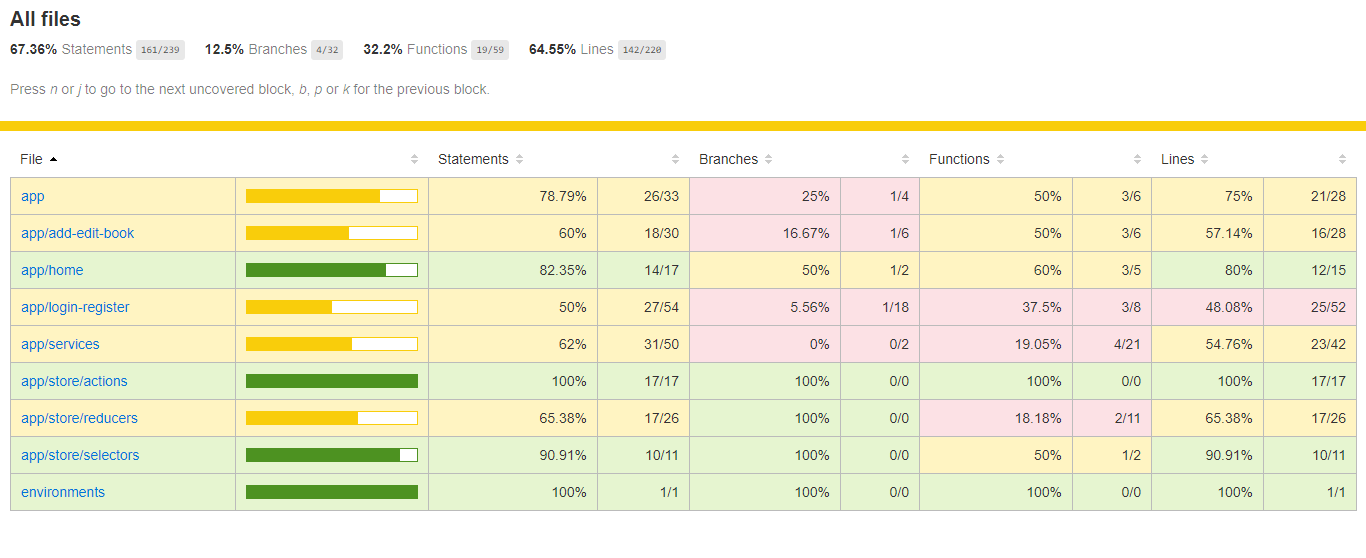
# **Tools and Libraries**

* Angular version - 10.1.5
* TypeScript - 4.0.2
* rxjs - 6.6.3
* ngrx/store – 10.0.1
* ngrx/entity – 10.0.1
* ngrx/effects – 10.0.1
* Jest – 26.6.0
* json-server-auth – 2.0.2
* @storybook – 6.0.26

# **Unit Testing**

**Total test cases: 18**

**Screenshots for unit test cases –**



# **Responsive**

* The application is responsive with all the device and desktop size.
* No Horizontal scrollbar for main application content.
* Used custom CSS for styling.

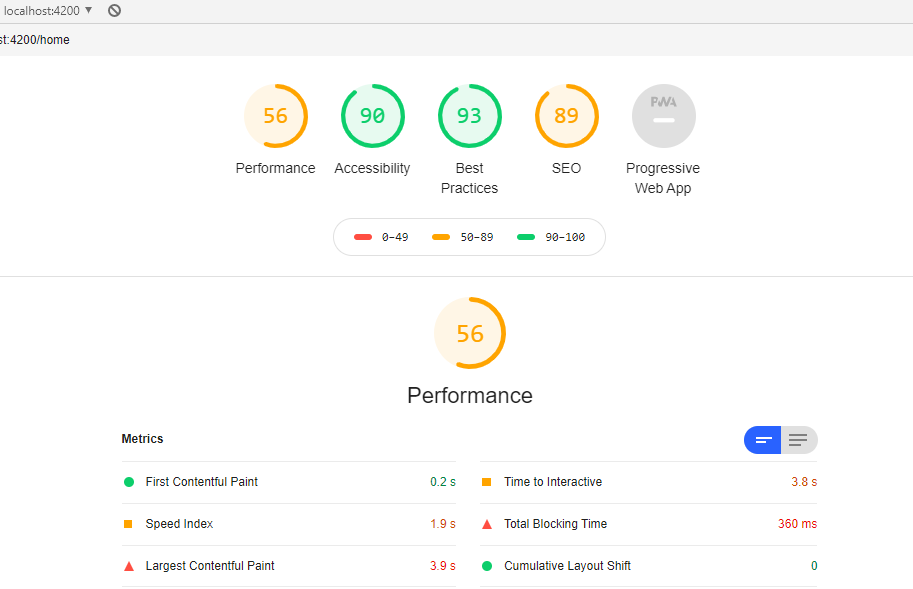
# **Accessibility**

For Accessibility, application is having appropriate HTML tags that are preferred for accessibility

User can run application via tab and enter.

# **Lighthouse**

Below is the Lighthouse report screenshot

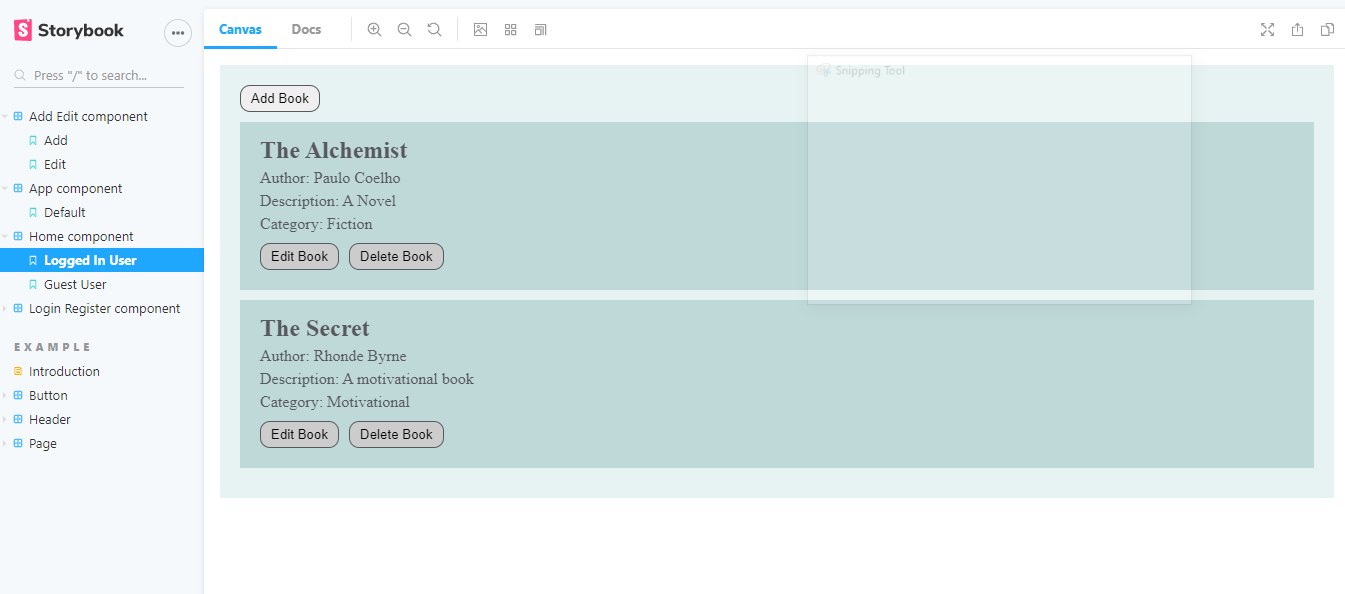


# **Story Book**

Storybooks are written for below components:

1. AppComponent
2. HomeComponent
3. LoginRegisterComponent
4. AddEditBookComponent

Below is the sample screenshot for Home - LoggedIn component



# **Setup**

1. Download and install node.js
2. Install npm and install @angular/cli
3. Git clone
4. Navigate to cloned folder & run **npm install**
5. To start application run **npm start**
6. For testing use **npm test**
7. For running storybook:

* First run the application using **npm** **start**
* Then run storybook **npm** **run** **storybook**